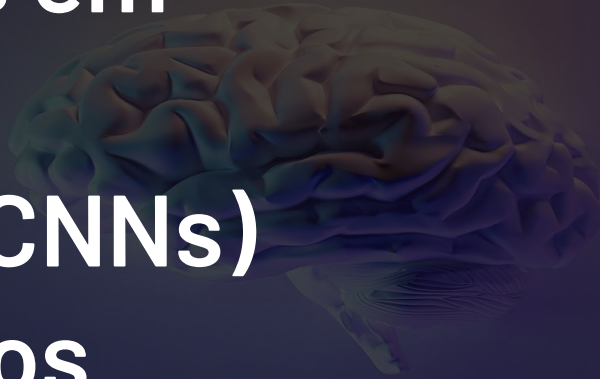
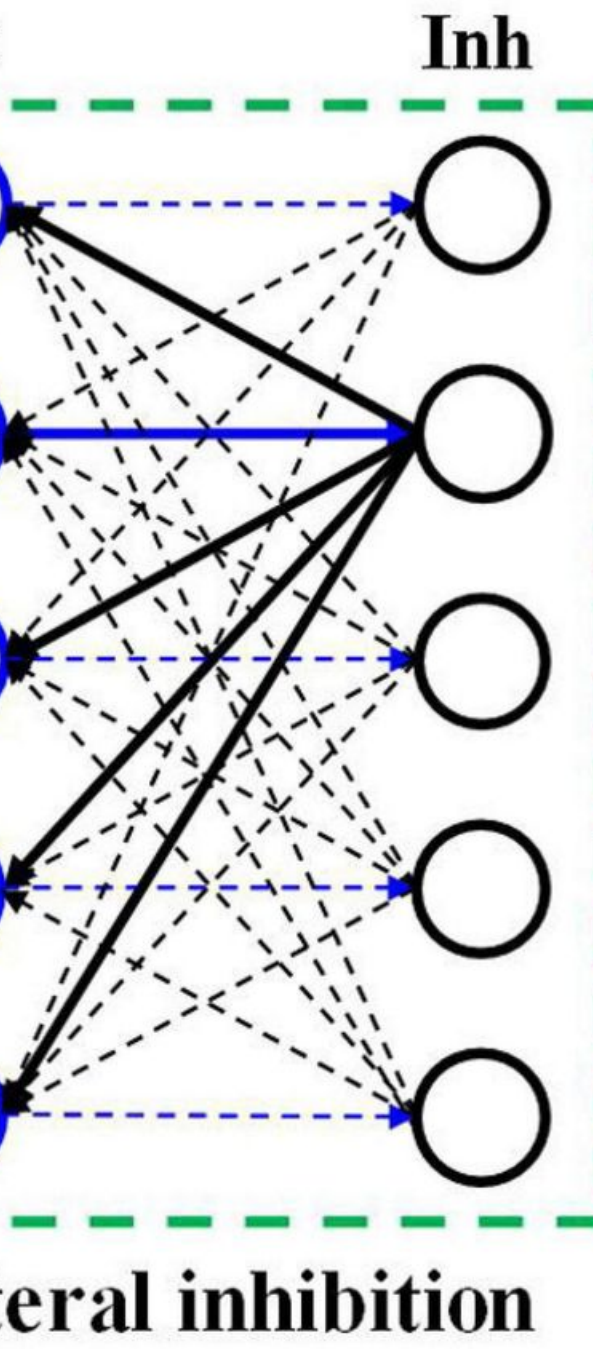


# Otimização de Hiperparâmetros em Redes Neurais Convolucionais (CNNs) usando Algoritmos Genéticos





# Referencial Teórico

1

Algoritmo Genético

2

Redes Neurais Convolucionais

1	0	0	0	0	0
---	---	---	---	---	---

1	0	1	1	0	0
---	---	---	---	---	---

0	1	1	1	1	0
---	---	---	---	---	---

0	1	0	0	1	0
---	---	---	---	---	---



# O que é algoritmo genético?

O algoritmo genético é um método de otimização inspirado no processo de seleção natural. Ele utiliza técnicas de reprodução, mutação e seleção para encontrar soluções eficientes para problemas complexos. É amplamente utilizado em áreas como inteligência artificial, otimização e aprendizado de máquina.

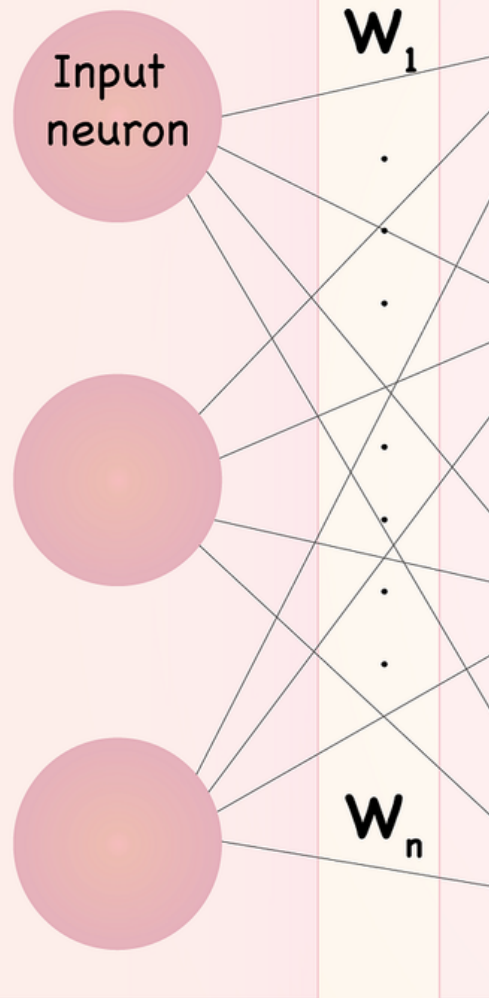


# O que é uma Rede Neural Convolucional (CNN)?

Uma Rede Neural Convolucional (CNN) é um tipo de rede neural projetada para processar dados com formato de grade, como imagens. Ela utiliza camadas convolucionais para extrair características relevantes das imagens e, em seguida, classificar ou fazer previsões com base nessas características. Um exemplo de aplicação de uma CNN é a classificação de imagens, onde a rede aprende a reconhecer objetos ou padrões em uma imagem.

Input layer

H



1 neuron per  
component

Ran

# Importância dos Hiperparâmetros

## 1 Papel Crucial

Os hiperparâmetros influenciam diretamente a eficácia do aprendizado de máquina.

## 2 Influência na Performance

Os hiperparâmetros afetam a velocidade e qualidade do processo de treinamento.

## 3 Otimização Manual

A otimização manual dos hiperparâmetros pode ser uma tarefa tediosa e desafiadora.



# Abordagem Proposta

## 1 Rede Neural Convolucional

Foco na otimização de uma CNN para o reconhecimento de imagens na escala de cinza.

## 2 Conjuntos de Dados

Experimentos conduzidos nos conjuntos de dados MNIST e Fashion-MNIST, utilizando TensorFlow.

## 3 Otimização Eficiente

O objetivo é maximizar a performance da CNN por meio da otimização eficiente dos hiperparâmetros.

## 4 Comparação

Comparar uma implementação própria do algoritmo genético com o eaSimple da biblioteca DEAP.



# Metodologia de Otimização

A rede será composta por 8 camadas:

1. A camada de entrada, que recebe  $28 \times 28 \times 1$  pixels, o tamanho das imagens em escala de cinza dos datasets.
2. Uma camada de convolução.
3. Uma camada de *pooling*.
4. Uma camada de convolução.
5. Uma camada de *pooling*.
6. Uma camada de *flatten*.
7. Uma camada densa.
8. A camada de saída, com 10 neurônios, um para cada classe.



Camada	Hiperparâmetro	Valores
Camada 2 (convolução)	Número de filtros	[32, 64, 128, 256]
	Tamanho dos <i>kernels</i>	[(3 × 3), (5 × 5), (7 × 7)]
	Função de ativação	[ReLU, GELU, <i>softmax</i> ]
Camada 3 ( <i>pooling</i> )	Tipo de <i>pooling</i>	[ <i>max pooling</i> , <i>average pooling</i> ]
	Tamanho do kernel	[(2 × 2), (3 × 3)]
Camada 4 (convolução)	Número de filtros	[8, 16, 32, 64]
	Tamanho dos <i>kernels</i>	[(3 × 3), (5 × 5), (7 × 7)]
	Função de ativação	[ReLU, GELU, <i>softmax</i> ]
Camada 5 ( <i>pooling</i> )	Tipo de <i>pooling</i>	[ <i>max pooling</i> , <i>average pooling</i> ]
	Tamanho do kernel	[(2 × 2), (3 × 3), (4 × 4)]
Camada 7 (densa)	Número de neurônios	[32, 64, 128, 256]
	Função de ativação	[ReLU, GELU, <i>softmax</i> , <i>sigmoid</i> ]
Camada 8 (saída)	Função de ativação	[ <i>softmax</i> , <i>sigmoid</i> ]

Tabela 1. Tabela mostrando os valores testados para cada hiperparâmetro otimizado.

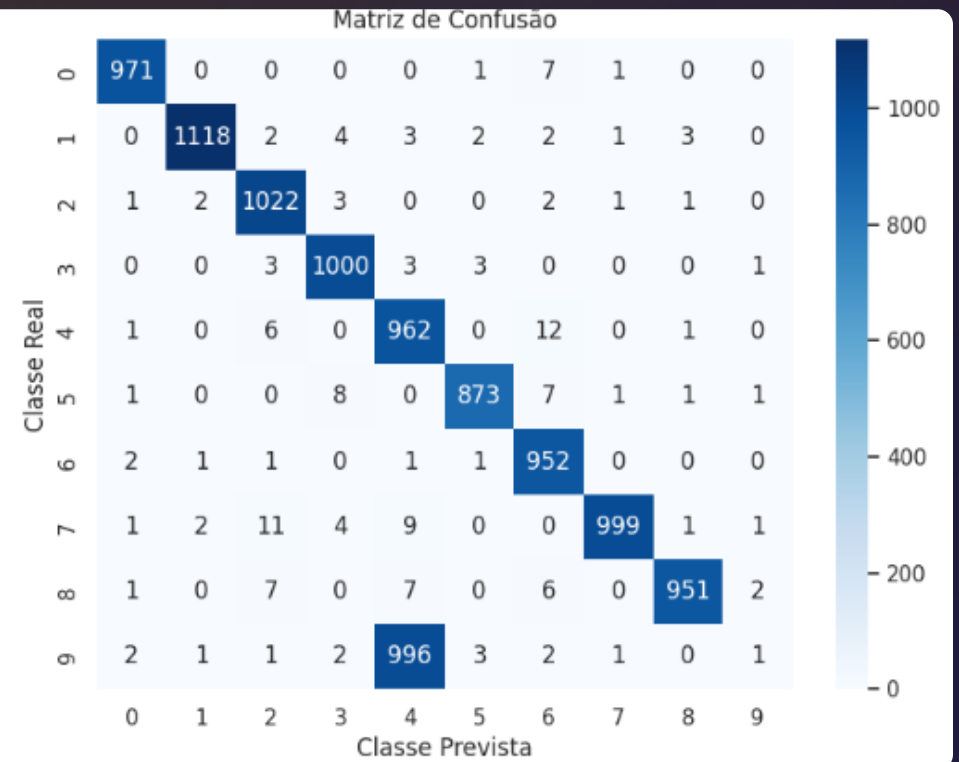
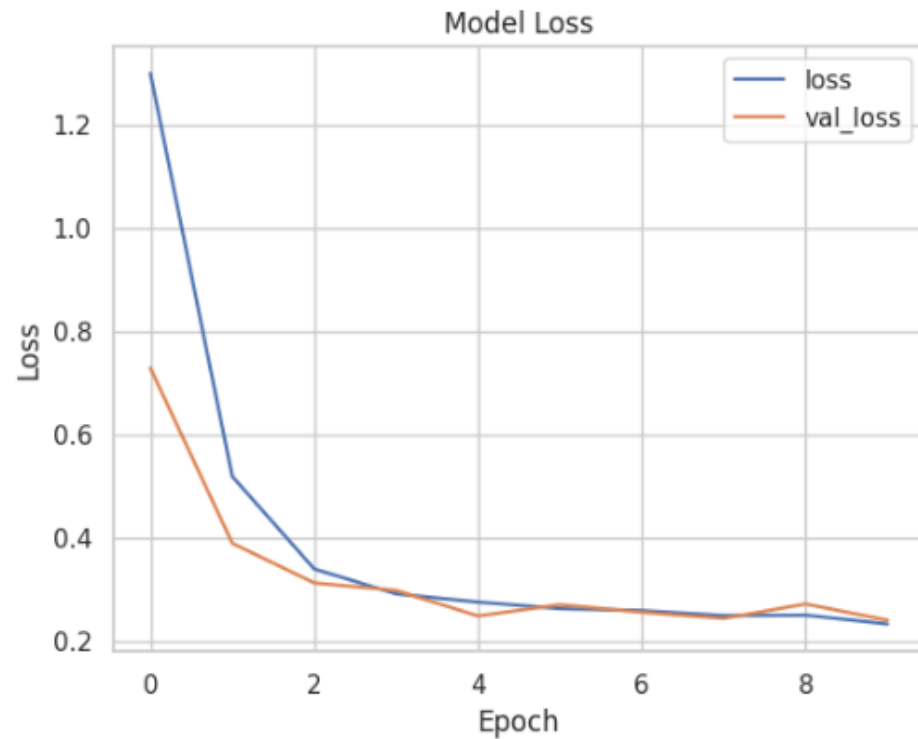
# Resultados

Camada	Hiperparâmetro	1	2	3	4
Camada 2 (convolução)	Número de filtros	64	64	219	81
	Tamanho dos <i>kernels</i>	(7, 7)	(7, 7)	(7, 7)	(7, 7)
	Função de ativação	GELU	GELU	GELU	ReLU
Camada 3 ( <i>pooling</i> )	Tipo de <i>pooling</i>	max	max	max	max
	Tamanho do kernel	(2, 2)	(2, 2)	(2, 2)	(2, 2)
Camada 4 (convolução)	Número de filtros	64	64	35	64
	Tamanho dos <i>kernels</i>	(5, 5)	(5, 5)	(7, 7)	(7, 7)
	Função de ativação	GELU	GELU	GELU	Softmax
Camada 5 ( <i>pooling</i> )	Tipo de <i>pooling</i>	max	max	avg	max
	Tamanho do kernel	(3, 3)	(3, 3)	(2, 2)	(2, 2)
Camada 7 (densa)	Número de neurônios	128	128	50	256
	Função de ativação	GELU	GELU	Softmax	ReLU
Camada 8 (saída)	Função de ativação	Sigmoid	Sigmoid	Softmax	Sigmoid

**Tabela 2.** Tabela mostrando os valores resultantes do algoritmo genético. 1 é o AG autoral para o MNIST, 2 é o AG autoral para o *Fashion* MNIST, 3 é o eaSimples para o MINST e 4 é eaSimples para o *Fashion* MNIST

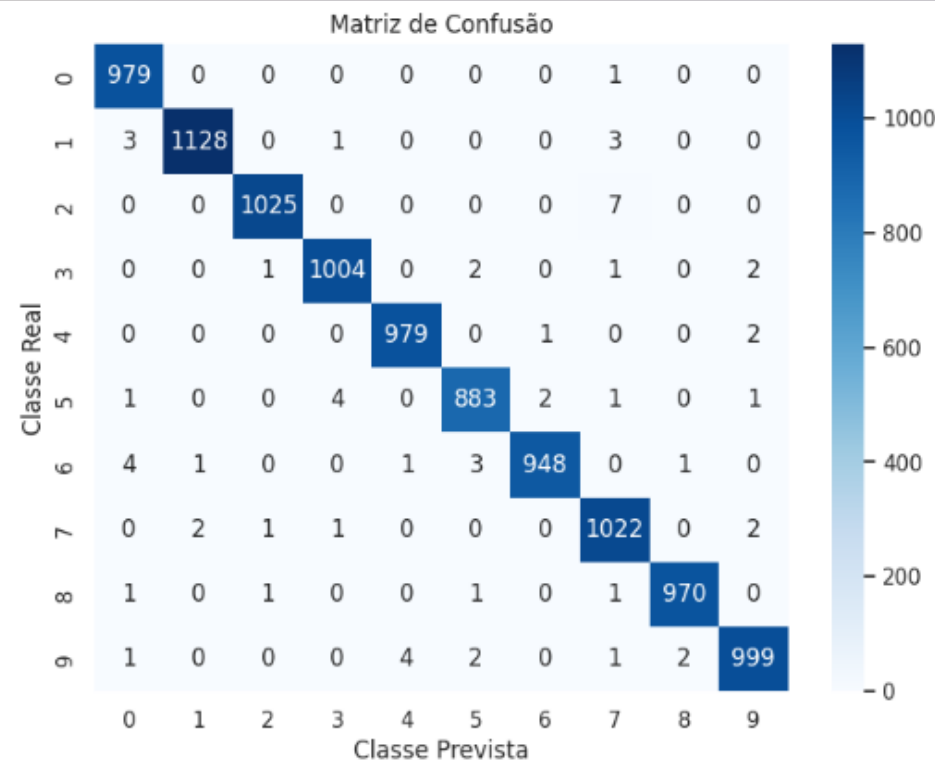
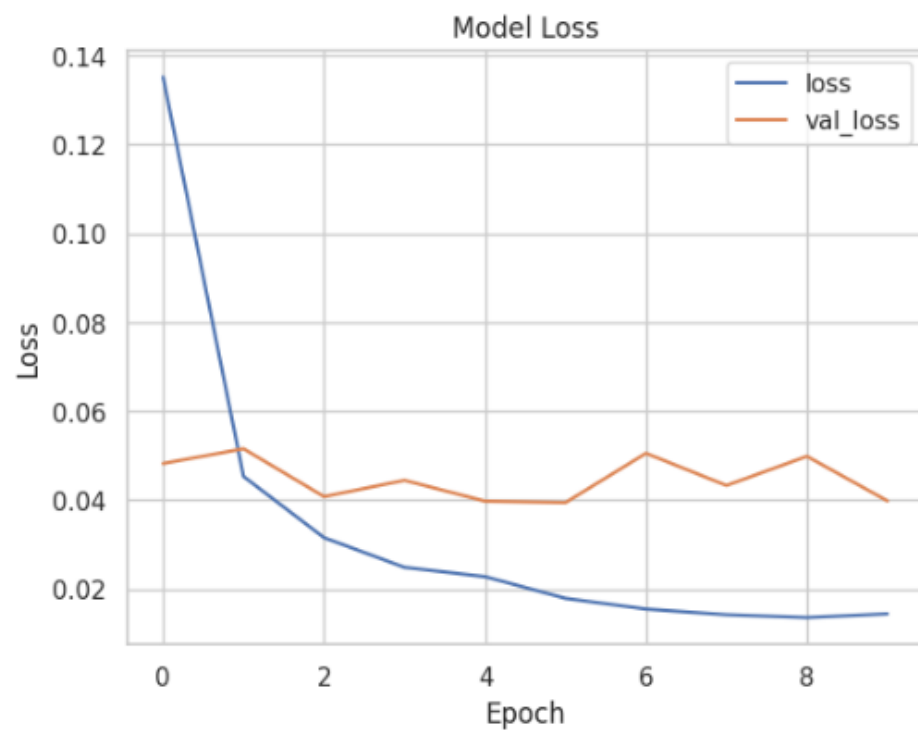


# MNIST - eaSimple



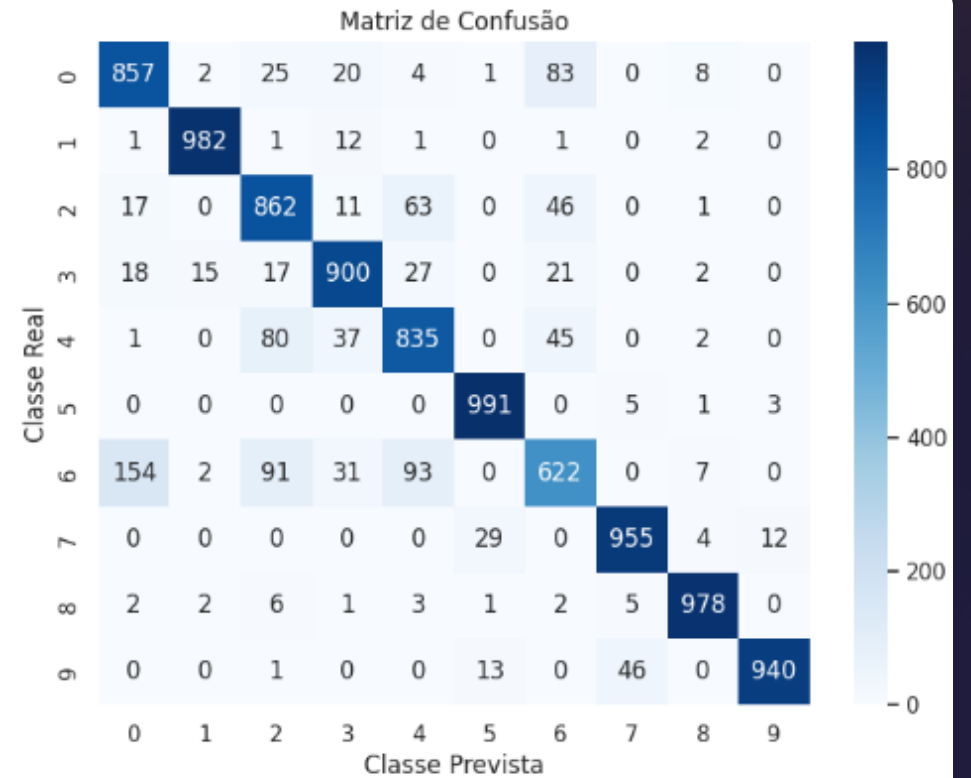
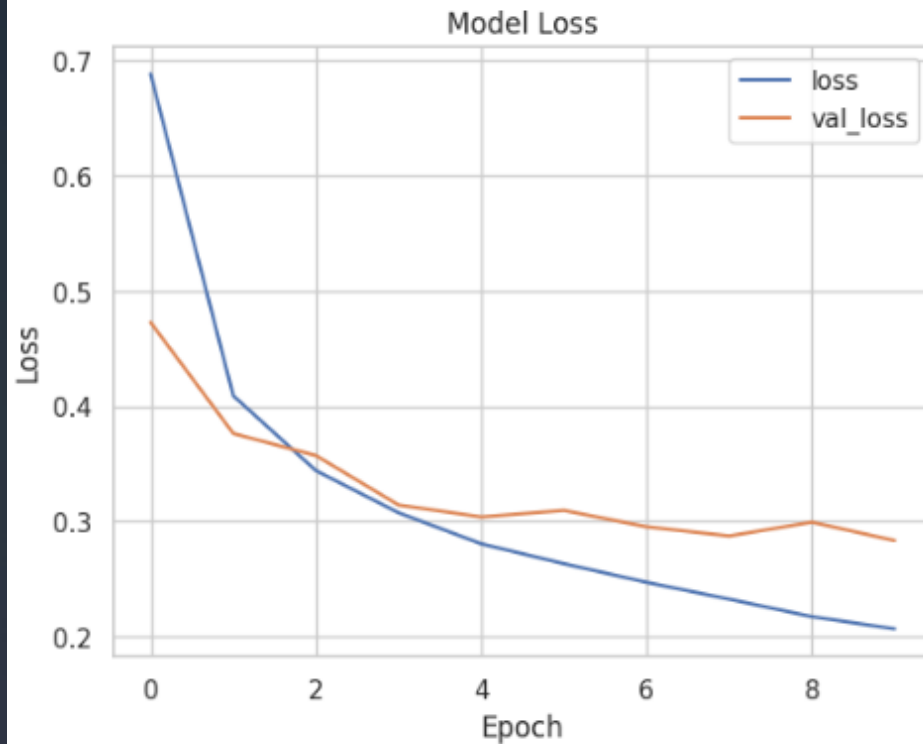
# MINST - Algoritmo Autoral

criada pelos autores.



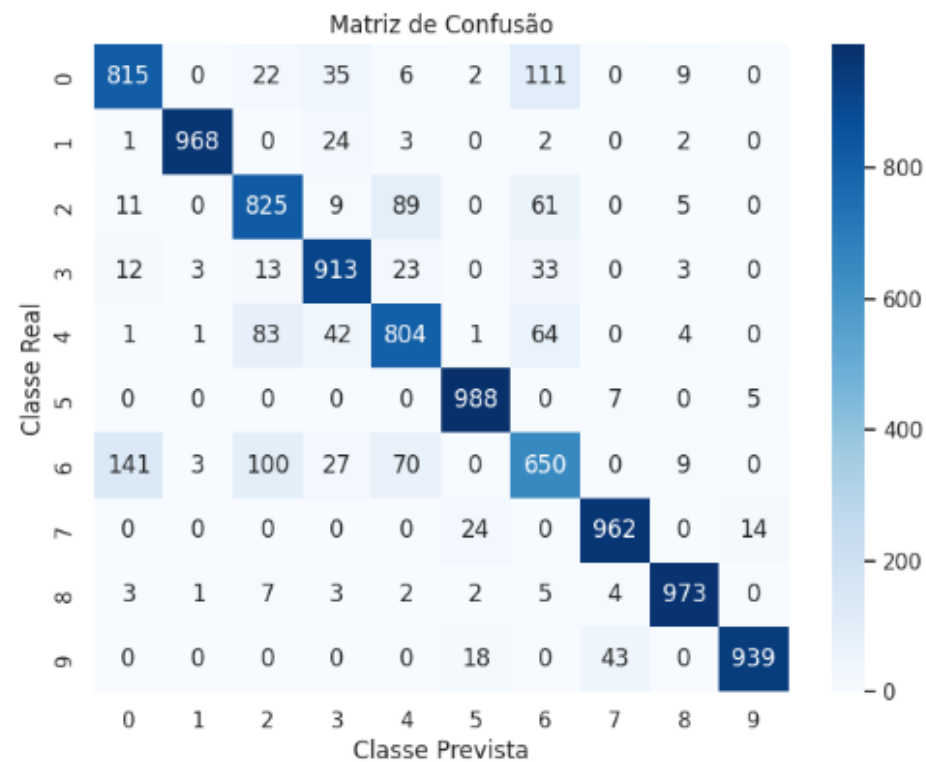
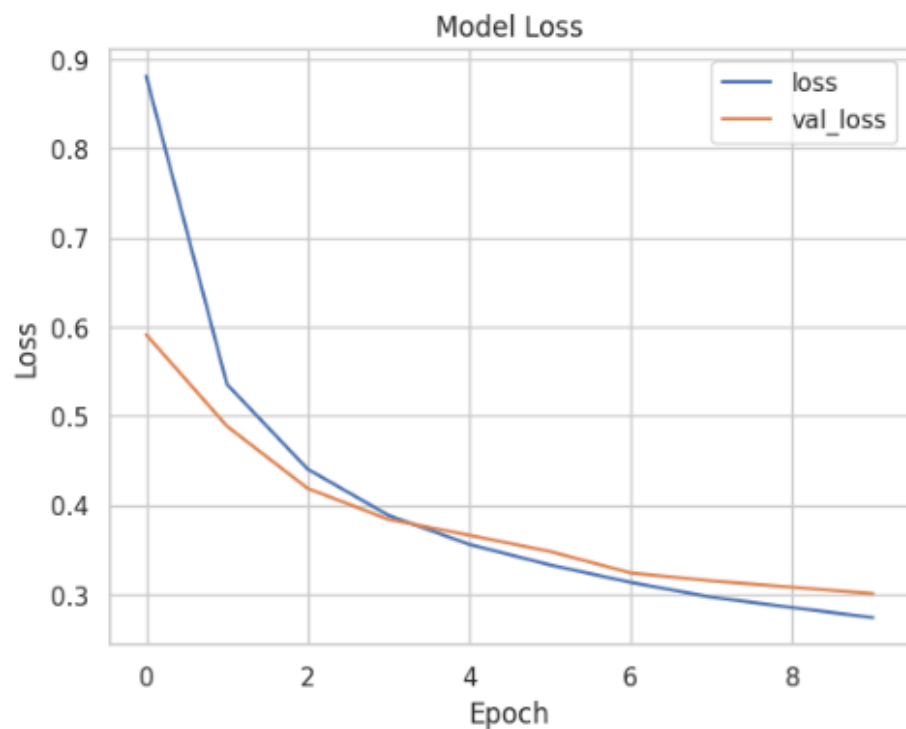
# Fashion MNIST - eaSimples

res.



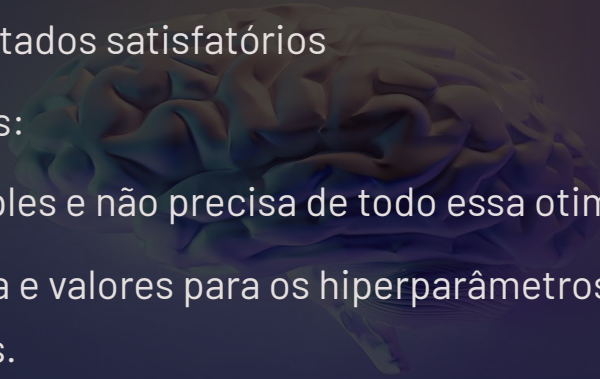
# Fashion MNIST – Algoritmo Autoral

gem elaborada pelos autores.

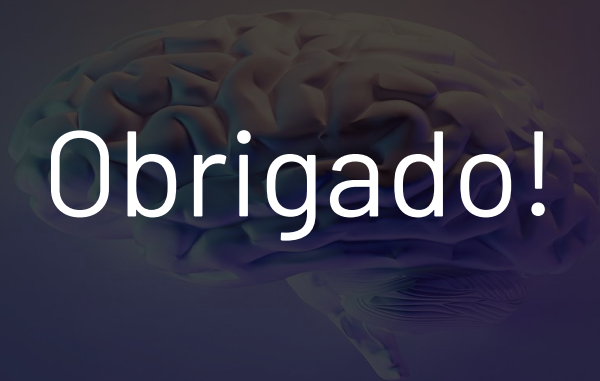


# Conclusão

- Ambos os métodos apresentaram resultados satisfatórios
- Porém, pela similaridade dos resultados:
  - Ou o problema é relativamente simples e não precisa de toda essa otimização em seus hiperparâmetros;
  - Ou, coincidentemente, a arquitetura e valores para os hiperparâmetros já eram suficientemente robustos para gerar esses resultados satisfatórios.
- Para trabalhos futuros, é interessante realizar análises mais aprofundadas para confirmar se os resultados obtidos foram coincidência ou não.







Obrigado!